

## A DSP-based real-time optical Doppler tomography system

Shikui Yan, Daqing Piao, Yueli Chen and Quing Zhu

Electrical & Computer Engineering Department, University of Connecticut, CT 06269-2157

**Abstract:** In this paper, we present a real-time data processing and display unit based on a custom designed digital signal processor (DSP) module for tissue structure and Doppler blood flow imaging. The DSP module is incorporated into a conventional optical coherent tomography (OCT) system. We also demonstrate the flexibility of embedding advanced Doppler processing algorithms in the DSP module. Two advanced velocity estimation algorithms previously introduced by us are incorporated in this DSP module. Experiments on intralipid flow demonstrate that this DSP system is capable of imaging pulsatile flow at a rate of several hundred pulses per minute. *In vivo* imaging capability with pulsatile blood flow is also demonstrated.

**OCIS codes:** (170.0170) Medical optics and biotechnology; (170.4500) Optical coherence tomography.

Correspondence: Prof. Q. Zhu, University of Connecticut, Storrs, CT 06269; e-mail [Zhu@engr.uconn.edu](mailto:Zhu@engr.uconn.edu); fax: 860-486-1273.

## 1. Introduction

Optical Doppler tomography (ODT) <sup>1, 2</sup> is a noninvasive, noncontact imaging modality that combines coherent gating and Doppler principle to obtain high-resolution cross-sectional imaging of tissue microstructure and blood flow, simultaneously. As a functional extension of optical coherence tomography (OCT), <sup>3</sup> ODT performs depth-resolved tomographic flow measurements in a series of time frames. Since ODT imaging is highly localized, the image acquisition speed must exceed the time scale of the movement of non-stationary objects in order to avoid motion artifacts. In addition, a real-time ODT data processing scheme providing rapid and continuous imaging and monitoring of blood flow synchronized with structural imaging is needed for timely diagnosis. The development of rapid-scanning optical delay line is the first technical challenge for real-time ODT, and it is practically solved. <sup>4, 5, 6</sup> The second issue is the development of real-time data processing and display unit. In general two steps of data processing are involved in simultaneous structure and flow velocity detection using ODT. First, a structural cross-sectional image is obtained by segmenting the raw interference data to obtain individual A-line profiles, and mapping the amplitude of each A-line profile to a series of fixed gray scales. Second, a flow image is obtained using Doppler analysis and displayed either aside or overlapped on top of the structural image for simultaneous visualization of flow regions and flow directions. The structural image formation is straightforward and fast in general by using Hilbert transform and envelope detection. The flow imaging however, requires intensive computations.

Because of ODT's promising potential in clinical applications, real-time ODT is under extensive investigations. Westphal et al. and Rollins et al. have employed autocorrelation scheme in

hardware to measure the phase of the correlation function, which corresponds to the local Doppler shift.<sup>7,8</sup> Zhao et al have proposed a method utilizing the polarization properties of light to obtain phase information and the method was implemented by optical Hilbert transformation.<sup>9,10</sup> Yang, et al. have implemented in-phase and quadrature (I & Q) demodulator in hardware and subsequent Kasai autocorrelation velocity estimator in software.<sup>11</sup> All these techniques focus on the hardware-implementation of a particular Doppler algorithm in order to eliminate the need of off-line processing of the recorded interference signals. The implemented Doppler algorithm, therefore determines the performance of the velocity estimator.

The Doppler analysis algorithms currently used in ODT can be classified into three major categories: a short-time Fourier transform (STFT)<sup>1</sup> method, a filtering method,<sup>12</sup> and a Hilbert transform<sup>13</sup> method. The short-time Fourier transform method extracts local Doppler shifts by measuring the spectrum centroid displacement of the interference signal. The filtering method directly maps the frequency shift at each pixel using digital filtering technique performed by a bank of narrow band-pass filters. The Hilbert transform method, also called correlation method, calculates the autocorrelation of individual A-scan line or cross-correlation between sequential A-scan lines, and estimates velocity changes from the phase of the correlation function. The STFT method is efficient in computation for its simplicity, but it is noisy. The filtering method is robust to noise and most accurate compared with STFT and Hilbert transform techniques, however its digital filtering process requires intensive computation. The Hilbert transform method is known to have highest velocity sensitivity when cross-correlation is taken between sequential A-lines. Nevertheless it is noisy at lower flow speed and/or lower signal-to-noise-ratio (SNR) region because correlation techniques are more sensitive to local decorrelation of the

flow field resulting from low SNR or high velocity gradient that is present at low flow velocity region.<sup>12, 14</sup> The Hilbert transform method is computational intensive due to the correlation process. An investigation on the off-line computation load of these three categories of algorithms reveals that the flow image processing of a typical data set of 512\*2048 requires approximately 15, 60, and 106 seconds for the three algorithm categories, respectively, on a Pentium III 800MHz-based PC.<sup>[12]</sup>

In the above-mentioned real-time ODT techniques primarily the Hilbert transform is implemented. Hilbert transform accomplished in hardware has the advantage of simple implementation, however, it lacks the flexibility when a different Doppler method is about to use. An ODT system based on DSP module, on the other hand, has the advantage of embedding any desired high-performance Doppler algorithm. In addition, it has the flexibility of incorporating other functional imaging algorithms such as spectroscopic OCT.<sup>15</sup> In this paper, we present a real-time ODT system using a low-cost digital signal processor (DSP) module embedded with two advanced Doppler algorithms. In this approach, a custom-designed DSP is incorporated into a conventional ODT system with a grating-based rapid scanning delay line. Configured with a signal processing evaluation module and an analog-to-digital (AD) daughterboard, this DSP performs data acquisition, structural imaging, and flow processing sequentially. DSP performance evaluation indicates the real-time data processing capability. Experiments on intralipid flow demonstrate that this DSP system is capable of imaging pulsatile flow at a rate of several hundred pulses per minute. *In vivo* blood flow imaging with pulsatile flow is also demonstrated. Two advanced velocity estimation algorithms, namely weight centroid

and digital filtering, are embedded in this DSP, and the superior performance of the sliding-window filtering algorithm is also demonstrated.

## 2. The flow velocity estimation algorithms embedded in DSP

In our previous work,<sup>12</sup> we discussed five velocity estimation algorithms, among which we introduced two new techniques: weighted centroid and sliding-window filtering. We also presented a quantitative assessment of the velocity estimation accuracies of these algorithms by simulations and *in vivo* study. These comparisons have shown that the weighted centroid technique slightly outperforms other three currently available algorithms. The sliding-window filtering technique is demonstrated to be superior to all other techniques on velocity estimation accuracy, most significantly at low signal-to-noise ratio (SNR). In terms of operational count, however, the computation time of sliding-window filtering method is 6 times higher than weighted centroid method. To take the advantage of the robust sliding-window filtering technique on flow estimation and also satisfy the real-time data processing demand, we have implemented a low-cost DSP-based signal processing module. This DSP module improves the calculation speed dramatically such that real-time data processing is readily achieved. In the following section, we give a brief review of these two algorithms that are embedded in the DSP.

Since ODT requires a broader bandwidth to detect frequency shifts due to moving scatterers, averaging of repeated data acquisition to improve SNR is necessary. By taking into account the averaging, we express the detected two-dimensional (2-D) interference signal after Hilbert transform as<sup>12</sup>

$$\tilde{z}_{k,i}(t) = |\tilde{z}_{k,i}(t)| e^{j\omega t} \quad (1)$$

where  $t$  is the time argument along the depth scanning direction,  $k$  represents the lateral dimension of the  $k$ th A-line,  $i$  is the index of the repeated A-line measurements, and  $\omega$  is the angular frequency of signal reflected from the target. The sampled interferometric signal is represented by  $\tilde{z}_{k,i}(t_n)$ , where  $t_n = nt_s$  and  $t_s$  is the sampling interval, and the power spectrum of  $\tilde{z}_{k,i}(t_n)$  is denoted with  $P^n_{k,i}(\omega)$ .

### 2.1 Weighted centroid algorithm

The weighted centroid algorithm, which is an extension of the conventional centroid method, is given by

$$\bar{\omega}_{k,i}(n) = \int_{-\infty}^{\infty} \frac{\omega [P^n_{k,i}(\omega)]^{\xi} d\omega}{[P^n_{k,i}(\omega)]^{\xi} d\omega} \quad (2)$$

where a positive integer  $\xi$  is introduced as the weight of the spectrum to emphasize the frequency component corresponding to the peak power in the spectrum.

### 2.2 Sliding-window filtering algorithm

The sliding-window filtering technique directly maps the frequency shift at each pixel using a digital band-pass filtering. In this approach, the detected signal  $\tilde{z}_{k,i}(t_n)$  at each depth scanning range  $t_n$  is processed by a filter bank consisting of  $M$  filters,  $f_1, f_2, \dots, f_M$ , yielding one signal at the output of each filter as illustrated in Fig. 1. This produces  $M$  output signals  $\tilde{O}_m(t_n)$ ,  $m = \{1, 2, \dots, M\}$  for each detected signal such that

$$\tilde{O}_m(t_n) = \tilde{z}_{k,i}(t_n) \otimes F^{-1}(f_m) \quad (3)$$

where the passband of  $f_m$  is  $[(2m-1)\pi/2M - \Delta\omega/2, (2m-1)\pi/2M + \Delta\omega/2]$  with  $\Delta\omega$  being the filter bandwidth. The output signals represent components of input signal that change at a rate corresponding to the passband of the filter. Then, at each range  $t_n$ , the energy in each filter,  $\varepsilon_m(t_n)$ , is estimated as <sup>[16]</sup>

$$\varepsilon_m(t_n) = \sum_{t'=t_n-\Delta t/2}^{t_n+\Delta t/2} [\tilde{O}_m(t')]^2 \quad (4)$$

where  $\Delta t$  represents a short range window centered at  $t_n$ . At any given  $t_n$ , the filter window position corresponding to the maximum  $\varepsilon_m(t_n)$ ,  $m = \{1, 2, \dots, M\}$ , represents the most significant frequency component, equivalently the frequency shift, of flow signal. The filtering window  $[\omega_1, \omega_2]$  determines the velocity resolution of this algorithm, and a better resolution requires a narrower filtering window. However, a robust estimate of the velocity requires a broader filtering window. A compromise between velocity resolution and robustness to noise has been achieved by using a bandwidth of  $\pi/32$  when a 2nd order Chebyshev filter of the first kind is chosen.

### 3. The architecture of DSP-based ODT system

The DSP-based ODT system is constructed on top of a conventional ODT device. <sup>17</sup> Functionally, this new system consists of four units (shown in Fig. 2): ODT scanner, control signal generator, embedded real-time processor, and image display module. The detailed function of each unit is described as follows.

#### Unit 1. ODT scanner

The ODT scanner is a typical balanced setup configured with one 1×2 and one 2×2 fiber couplers (see Fig.3). The low coherence source is a superluminescent diode with 1300 nm center wavelength, 40 nm spectral width, and 2.0 mW maximum output power. In the reference arm, a scanning optical delay line based on Littrow-mounting of diffraction grating is used to generate range scanning and a phase modulation for carrier frequency as well. <sup>[12]</sup> In the sample arm, a galvanometer and an achromatic lens are integrated to perform repeatable lateral scanning that is essential for continuous monitoring of flow. In the detection arm, the signal is detected differentially by an auto-balanced receiver, with which the source intensity noise is rejected. The signal is then amplified, bandpass filtered and digitized.

Using this system, cross-sectional imaging of one frame per second is obtained by driving reference arm and sample arm galvanometers with 128 Hz triangle wave and 1Hz sawtooth wave, respectively. The reference galvanometer is capable of scanning at much higher frequency, however, 128 Hz scan speed is used due to the bandwidth limitation of the dual-balanced receiver. Each frame consists of 128 axial scans containing 2048 data points at each round of scan. Because we used 50% duty cycle of galvanometer scanning, each A-line covers 1024 points, and 4 consecutive A-lines are then averaged to improve the SNR. In the axial dimension, 16 points are averaged at one pixel, resulting pixel dimension of 64×32 covering an area of 2.5mm×1.0mm (intralipid study) and 2.5mm×2.0mm (rat study).

## Unit 2. Control signal generator

To synchronize the axial scanning, lateral scanning as well as data acquisition, we have built a control signal generator based on a National Instruments (NI) multi-function Data Acquisition



(DAQ) card, PCI-MIO-16E-1. As described in Fig. 4, this unit generates synchronization signals for ODT scan and data acquisition: one fast triangle wave for reference arm scanning and one slow sawtooth wave for lateral scanning as previously mentioned, and a TTL-compatible triggering signal for data acquisition and real-time processing.

### Unit 3. Embedded real-time processor

Shown in Fig. 5 (a) is the newly embedded real-time module that is a combination of TMS320C6701 DSP Evaluation Module ('C6701 EVM, Texas Instruments Inc.)<sup>18-20</sup> and analog expansion daughterboard AED-109<sup>21</sup> (Signalware Inc.). The 'C6701 EVM board is a complete DSP system based on 'C6701 DSP chip operating up to 1336 million instructions per second (MIPS) with a CPU clock rate of 167 MHz. With 'C6701 one can actually perform two multiplication and accumulation operations within a single clock cycle. This board also provides 256 KB of external 133-MHz synchronous burst static RAM (SBSRAM) and 8 MB of external 100-MHz synchronous dynamic RAM (SDRAM), both of which could be used as data buffers. The daughter board AED-109 has two dual-channel 12-bit A/D converters with up to 8-MHz sampling speed and two D/A converters; the input channels have been configured into differential inputs for higher SNR consideration. In addition to analog channels, 16-buffered digital signals can be used as flags for system control.

The data flow with this real-time processor is shown in Fig. 6. When ODT scanning starts, the daughter board AED-109 acquires the signal from the ODT scanner, and the 12-bit data is buffered into 256-word two-way FIFO in the field programmable gate array (FPGA) XCV100E6 (Xilinx Inc.). As soon as this FIFO is full, XCV100E6 triggers 'C6701 DSP to transfer data to

'C6701 in direct memory access (DMA) mode. The circular buffer in the 'C6701 DSP memory, which is segmented into 8 frames with 256 words per frame, can hold 2K words data acquired during one complete fast scanning. The transferred 256-word data is saved into one frame of circular buffer, then the circular buffer is polled to process the new coming data. For structural image formation and flow calculation using weighted centroid algorithm, this DSP module processes each section (256 words) in the circular buffer continuously; however, for flow calculation using sliding-window filtering algorithm, DSP needs to acquire 4 continuous frames before it starts to process 1024 data points for a complete A-scan line. The processed structural image and flow image data are saved into each buffer in the SDRAM on 'C6701 EVM. The next step is to load image data into PC for display.

#### Unit 4. Image display module

A real-time image display is the last, yet critical step of this real-time DSP-ODT system. The real-time data exchange (RTDX) provided by Texas Instruments Inc. offers continuous bi-directional data exchange in real time with minimal perturbation to the application. However, in 'C6701 EVM, the maximum RTDX rate is 20 Kbytes/second. Since structural image and flow image data are in floating data format (4 bytes), a total of 96 Kbytes data needs to be loaded in one image frame (1 second) for simultaneous display of both structural and flow images, which is beyond the RTDX speed of our current DSP. To overcome this problem, the host-port interface (HPI) between DSP and PC is employed to load image data right after processing. With this technique, flow image processing and image uploading are performed alternately. During the operation DSP is continuously processing and saving image data into 8 Mbytes SDRAM on 'C6701 EVM to avoid data loss. After DSP has completed desired frames, all of the image data

are loaded at one time. This strategy assures the image acquisition speed up to a maximum of 128 A-scans per frame, which is the actual scan speed. If a photo-receiver with broader bandwidth is employed in ODT scanner to accommodate faster scanning, and a new high-speed RTDX (HS-RTDX) with 2 Mbytes/second transfer rate is implemented, we expect to achieve much higher processing and display updating speed than current 1 Hz frame rate.

#### **4. Evaluation of DSP performance**

To study the performance of the introduced DSP, we estimated the DSP cycles of three processing modes: structural image, weighted centroid flow image and sliding-window filtering flow image. The resulting operational structures for implementing each of them are shown in Fig. 7 and are described in the following paragraphs.

For structural image, one set of A-scan line with 1024 data points is segmented uniformly into 64 sections, with 16 points per section. The data in each section is then Hilbert transformed by floating-point FFT and IFFT pair. For each section, the maximum of the absolute values of Hilbert transformed data is determined, which represents the signal level in the local pixel. Therefore, a total of 64 pixel values are obtained for single A-scan data in structural imaging. The most calculations are FFT and IFFT pairs, resulting 0.043M cycles for fixed-point calculation and 0.041M cycles for floating-point calculation.

For weighted centroid algorithm, again, one set of A-scan line is segmented into 64 sections. The data in each section is zero-padded to 64 points, and then Fourier transformed by 64-point complex floating-point FFT. From the calculated 32-point spectrum value of each section, the

Doppler frequency shift is calculated by finding the local center frequency with weighted centroid principle and subtracting the known modulation frequency. Again, a total of 64 Doppler shift values are obtained for single A-scan data in flow imaging by weighted centroid technique. The most calculations focus on FFT. By using complex radix-2 floating point FFT, about 0.096M cycles are needed to finish the calculation of one 1024-point A-scan data. If complex radix-4 fixed point FFT is implemented instead, only 0.077M cycles are needed.

For sliding-window filtering algorithm, a complete set of A-scan line with 1024 data points is filtered individually by 32, 2nd order IIR bandpass filters, each of which has one of 32 evenly distributed narrow passbands from 0 to  $\pi$ , and the passbands of all filters cover the entire spectrum. The filtered 1024 data is segmented into 64 sections equally, with 16 points per section. In each section, the filter number corresponding to the maximum signal power after these 32 filters is indexed as the center frequency, and the Doppler shift is calculated by subtracting the modulation frequency. The resulted total number of Doppler shift values for single A-scan data in sliding-window filtering method is 64. The most calculations are IIR filtering banks. The execution takes about 0.32M cycles for 1024-point fixed-point calculation, and 0.61M cycles for floating point calculation. As listed in Table 1, the most intensive calculation is on sliding-window filtering technique, which is about 6 times as that of weighted centroid technique. Nevertheless, for one A-line at a duration of 1/128 second, the computation load of all three modes are within the 1.3M cycles real-time limitation of the 167MHz 'C6701 EVM.

The above-described DSP speed estimation is based on the approximation that the DSP has enough internal memory. However, 'C6701 DSP provides only 64 Kbytes internal memory for program storage and 64 Kbytes internal memory for data storage. As a result, the relatively low speed external memory in SBSRAM and SDRAM has to be used for DSP object file because it has a size of 95 Kbytes, which exceeds the 64 Kbytes internal program memory. If a code is present in an external memory, more cycles are demanded for its execution. In addition, a large amount of data buffers are needed in external memory. By optimizing memory allocation and code efficiency, we have achieved continuous calculation for simultaneous structural and flow imaging in less than 1 second with the current DSP module.

To verify that our DSP code is efficient, we have evaluated the ratio of DSP speed for two flow processing modes and compared it with Matlab calculations. In Matlab, 2.5 seconds and 17 seconds are needed for weighted centroid technique and sliding-window filtering technique, respectively, on a Pentium-4 2.0GHz PC. The ratio of speed in Matlab is 6.8, which is very close to 6.4 achieved by 'C6701 DSP using floating-point calculation.

'C6701 is a floating point DSP, and it can operate in either fixed-point or floating-point calculation. 'C6701 uses 2-byte short data type in fixed-point FFT and IIR functions; and 4-byte floating data type in floating-point FFT and IIR functions. Because Matlab uses 8-byte double data type to do the calculation, we can use the result from Matlab as a benchmark, and the calculation accuracy of fixed-point and floating-point can be found by studying the mean-square-error (MSE) with respect to the Matlab results. Table 2 lists the MSE in DSP calculation for fixed-point and floating-point data types. By comparing Tables 1 and 2, we can see that there is

a trade-off between speed and calculation accuracy on choosing either fixed-point or floating-point DSP. Since our DSP system is able to process flow information in desired time frame, the floating-point DSP is favorable for its higher accuracy and is therefore implemented in our code.

## 5. Results

### 5.1 Experimental flow

To demonstrate the performance of our DSP-ODT system, we constructed experimental flow setup with 0.5% intralipid solution. The ODT probing-beam is focused to the axis of a glass capillary tubing with 1.0mm inner diameter, 1.2mm outside diameter and 100mm in length. The glass capillary is mounted to a rotary stage, and connected to a 2-meter long 3/32" plastic tubing. The mean flow velocity of intralipid through the glass capillary tubing is converted from the time interval of liquid traveling 1 meter distance in the 3/32" plastic tubing that is placed straightly on a horizontal plane. A peristaltic tubing pump is used to drive the intralipid through the capillary tubing and circulate it in the tubing loop. The pulsation induced by peristaltic pumping is used to emulate the *in vivo* blood circulation. The tubing pump gives approximately 3 pulses per second at its minimum rotation speed.

Using this setup, 30 frames are taken continuously at 1Hz frame rate. During this 30 frames interval, the pumping speed has been manually changed, with a sequence of no flow, minimum speed, gradually increased speed, gradually decreased speed, increasing/decreasing again, and stop. One frame of the real-time acquired and processed flow images is shown in Fig.8. In those flow images there are several color strips distributed perpendicular to the lateral\_scanning direction in the intralipid flow region. The axial and transverse flow profiles are also plotted,

where the axial profile gives clear laminar flow pattern within single color strip corresponding to one pulse of flow, and the transverse profiles reveal the pulsation of the flow. The number of flow cycles and the magnitude of frequency shift correspond very well with the numbers of pulses generated by the pump and the speed of the flow. With this DSP-ODT system, we have captured intralipid flow of 10 pulses per second. This suggests that our system is capable of monitoring pulsatile flow circulation of human and experimental animals.

To investigate the performance of the two embedded Doppler algorithms in this real-time processing scenario, we compared the flow velocity estimation by weighted centroid technique and sliding-window filtering technique at 3 pulses per second (PPS) and 6 PPS (shown in Fig. 9). It is observed that the sliding window filtering method gives much less noisy, more accurate flow profile, and more clear pulsation pattern than weighted centroid algorithms does. This result is consistent with our simulations and experiments reported in Ref. 12.

### 5.2 *In vivo* study

We have conducted *in vivo* study on animal to validate the real-time imaging capability of our DSP-based ODT system. A 78-gram rat was prepared and operated under the protocol approved by the University of Connecticut (Z2400201). The blood vessels imaged were inside the opened abdomen, and the animal was euthanatized at the end of the study. 30 Frames are also taken from this rat sample, and one frame of the processed structural and flow images are shown in Figs 10. In Fig.10, the two subsurface bi-directional blood vessels are clearly identifiable. Since small blood vessels were imaged inside abdomen cavity, the color strips related to the pulsatile flow seen in intralipid experiment were not observed. This *in vivo* study indeed demonstrates

that our system is capable of monitoring *in vivo* bi-directional blood flows with reasonable sensitivity in real-time.

## 6. Discussions and Summary

In our DSP based ODT system, we have the flexibility of modifying system parameters that can be easily done through front panel of Labview software. The easily adjustable parameters include scanning speed of the galvanometer, the threshold in flow processing, number of averages, etc. We also have the flexibility of incorporating different flow algorithms in DSP processing module because the client program in DSP has been modularized into DAQ module, processing module and communication module. It is easy to implement new algorithms into processing module without changing the DSP code structure. Instead of implementing the DSP code in hardware, we have host program executed by Labview, which downloads DSP object file to 'C6701 EVM before the ODT system starts, and triggers DSP after initializing DSP-ODT system parameters. Therefore, any new modifications on algorithms can be easily implemented in software. The performance of this DSP system can be further improved if DSP board with higher system clock or multiple DSP boards can be employed.

In summary, we have designed a DSP-based ODT module features real-time data acquisition and processing by embedding two advanced Doppler analysis algorithms. This system incorporates a low-cost custom-designed DSP module into a conventional ODT system. Experiments of Intralipid flow and *in vivo* study have demonstrated the real-time capability of this system in capturing and monitoring pulsatile flows.



## **7. Acknowledgments**

The authors acknowledge partial funding support from National Institutes of Health (NIH 1R01 DE11154-03). Dan Schwartz from the Animal Services of University of Connecticut is greatly appreciated for animal preparation. Shikui Yan and Quing Zhu's email addresses are [skyan@engr.uconn.edu](mailto:skyan@engr.uconn.edu); [zhu@engr.uconn.edu](mailto:zhu@engr.uconn.edu);

## References:

1. Z.P. Chen, Y.H. Zhao, S.M. Srinivas, J.S. Nelson, N. Prakash, and R.D. Frostig, "Optical Doppler tomography," IEEE J. Sel. Top. Quant. Elect., **5**, 1134-1142 (1999).
2. J.A. Izatt, M.D. Kulkarni, S. Yazdanfar, J.K. Barton and A.J. Welch, "*In vivo* bi-directional color Doppler flow imaging of picoliter blood volumes using optical coherence tomography", Opt. Lett., **22**, 1439-1441 (1997).
3. D. Huang, E.A. Swanson, C.P. Lin, J.S. Schuman, W.G. Stinson, W. Chang, M.R. Hee, T. Flotte, K. Gregory, C.A. Puliafito, and J.G. Fujimoto, "Optical coherence tomography," Science, **254**, 1178-1181 (1991).
4. G.J. Tearney, B.E. Bouma, and J.G. Fujimoto, "High-speed phase- and group-delay scanning with a grating-based phase control delay line", Opt. Lett., **22**, 1811-1813 (1997).
5. A.M. Rollins, M.D. Kulkarni, S. Yazdanfar, R. Ung-arunyawee, and J.A. Izatt, "*In vivo* video rate optical coherence tomography", Opt. Expr., **3**, 219-229 (1998).
6. N.G. Chen, and Q. Zhu, "Rotary mirror array for high-speed optical coherence tomography," Opt. Lett., **27**, 607-609 (2002).
7. V. Westphal, S. Yazdanfar, A.M. Rollins, and J.A. Izatt, "Real-time, high velocity-resolution color Doppler optical coherence tomography," Opt. Lett., **27**, 34-36 (2002).
8. A.M. Rollins, S. Yazdanfar, J.K. Barton, and J.A. Izatt, "Real-time *in vivo* color Doppler optical coherence tomography," J. Biomed. Opt., **7**, 123-129 (2002).
9. Y. Zhao, Z. Chen, Z. Ding, H. Ren, and J.S. Nelson, "Real-time phase-resolved functional optical coherence tomography by use of optical Hilbert transformation," Opt. Lett., **27**, 98-100 (2002)

10. Z. Ding, Y. Zhao, H. Ren, J.S. Nelson, and Z. chen, "Real-time phase-resolved optical coherence tomography and optical Doppler tomography," *Opt. Exp.*, **10**, 236-245 (2002).
11. V.X.D. Yang, M.L. Gordon, A. Mok, Y. Zhao, Z. Chen, R.S.C. Cobbold, B.C. Wilson, and I.A. Vitkin, "Improved phase-resolved optical Doppler tomography using the Kasai velocity estimator and histogram segmentation," *Opt. Commun.*, **208**, 209-214 (2002).
12. D. Piao, L. Otis, N.K. Dutta, and Q. Zhu, "Quantitative assessment of flow velocity estimation algorithms for optical Doppler tomography imaging," *Appl. Opt.*, **41**, 6118-6127 (2002).
13. Y.H. Zhao, Z.P. Chen, C. Saxer, S.H. Xiang, J.F. de Boer, and J.S. Nelson, "Phase-resolved optical coherence tomography and optical Doppler tomography for imaging blood flow in human skin with fast scanning speed and high velocity sensitivity", *Opt. Lett.*, **25**, 114-116 (2000).
14. P.C. Li, C.J. Cheng, and C.K. Yeh, "On velocity estimation using speckle decorrelation", *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, **48**, 1084-1091 (2001).
15. Jian Ye, A.W. Schaefer, D.L. Marks, J.J. Reynolds, and S.A. Boppart, "Real-time processing of spectroscopic optical coherence tomography," *Proceedings of IEEE International Symposium on Biomedical Imaging*, TP-P1.22, Washington D.C., July 2002
16. J.R. Crowe, B.M. Shapo, D.N. Stephens, D. Bleam, M.J. Eberle, E.I. Cèspeles, C.C. Wu, D.W.M. Muller, J.A. Kovach, R.J. Lederman, and M. O'Donnell, "Blood speed imaging with an intraluminal array," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, **47**, 672-681 (2000).
17. D. Piao, and Q. Zhu, "Quantitation of Doppler angle and flow velocity mapping by combining Doppler shift and Doppler bandwidth measurements in optical Doppler tomography," *Appl. Opt.*, accepted, 2003.

18. Rulph Chassaing, "DSP Application Using C and the TMS320C6x DSK", John Wiley & Sons, Inc. 2002.
19. Texas Instrument, Inc., TMS320C62x DSP library Programmer's Reference, April 2002.
20. Texas Instrument, Inc., TMS320C67x FastRTS library Programmer's Reference, Oct 2002.
21. Signalware Co., "Documentation Package for AED-109 Multi-Channel Analog Expansion Daughterboard", Ver.1.0, Oct. 2001

Table 1. Calculation cycle evaluation of 3 processing modes based on 167MHz 'C6701 EVM.

Processing Mode	Fixed-point	Floating-point
Structural	0.043M cycles (0.26ms *)	0.041M cycles (0.25ms *)
Weighted Centroid	0.077M cycles (0.47 ms *)	0.096M cycles (0.58 ms *)
Sliding-window Filtering	0.32M cycles (1.9 ms *)	0.61M cycles (3.7 ms *)

Note: \* are calculated with 167MHz system clock of TMS320C6701 EVM

Table 2. DSP calculation accuracy of fixed-point and floating point compared with Matlab calculation in 8 bytes double data format. The number is the mean square error with respect to Matlab results.

Processing Mode	Fixed-point (2-byte short)	Floating-point (4-byte float)
Structural	$3.5 \times 10^{-2}$	$1.5 \times 10^{-6}$
Weighted Centroid	$6.8 \times 10^{-3}$	$3.4 \times 10^{-7}$
Sliding-window Filtering	$3.9 \times 10^{-2}$	$6.1 \times 10^{-4}$

## **Figure captions**

Fig. 1. Schematic of sliding-window filtering technique. The sliding-window filtering bank is applied to measure the components of each interference signal that change at a rate corresponding to the passband of the filters.

Fig. 2. Schematic of the DSP-based ODT system.

Fig. 3. ODT scanner, which is a typical balanced setup with both axial and lateral scanning, performed by galvanometer. O1, O2, microscope objectives; G1, G2, galvanometers; L1, L2, achromatic lenses.

Fig. 4. The synchronization strategy of ODT scanning and data acquisition.

Fig. 5. Photograph of C6701 EVM board with AED-109 daughter board plugged in.

Fig. 6. Data flow in DSP-ODT system: hardware prospective.

Fig. 7. Code structures of 3 processing modes: (a) Structural imaging; (b) Weighted centroid flow imaging; (c) Sliding-window filtering flow imaging.

Fig. 8. Experimental results on pulsatile intralipid flow. (a) 3 pulses per second; (b) 6 pulses per second.

Fig. 9. Axial and transversal velocity A-scan profiles by weighted centroid algorithm and sliding-window filtering algorithm for pulsatile intralipid flow. (a) 3 pulses per second; (b) 6 pulses per second.

Fig. 10. In vivo blood flow imaging of the abdominal blood vessels of a small rat. Two bi-directional blood vessels are clearly identified. (a) Weighted centroid flow image superimposed on structural image; (b) Sliding-window filtering flow image superimposed on structural image.



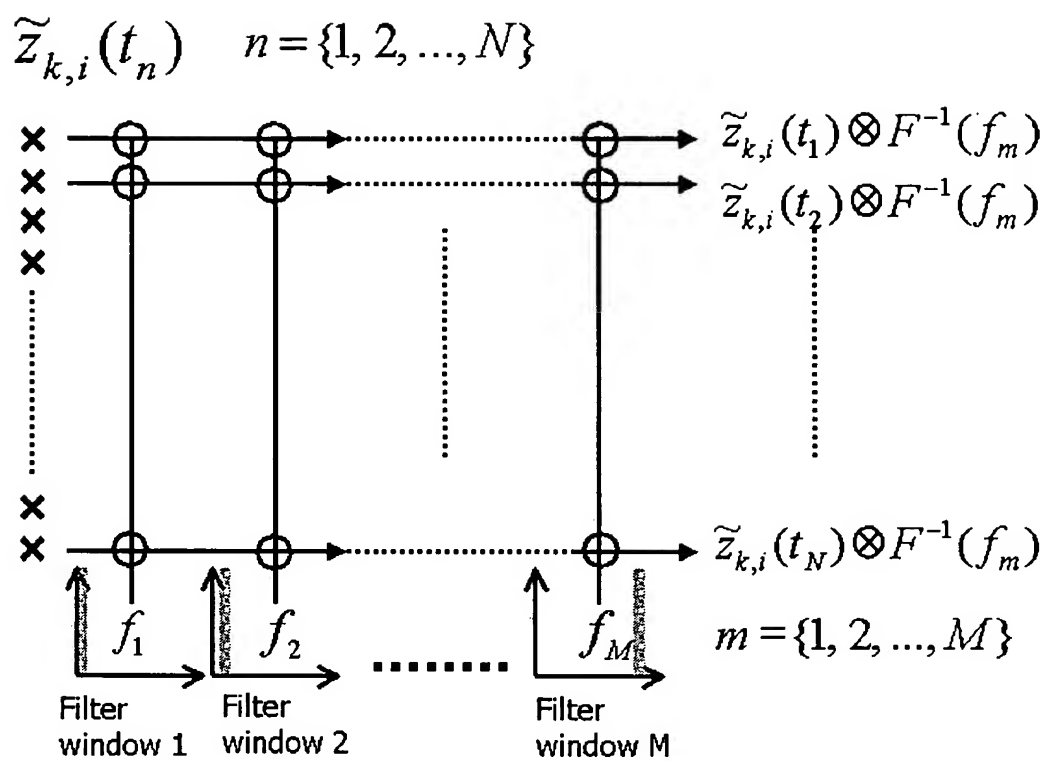


Fig. 1

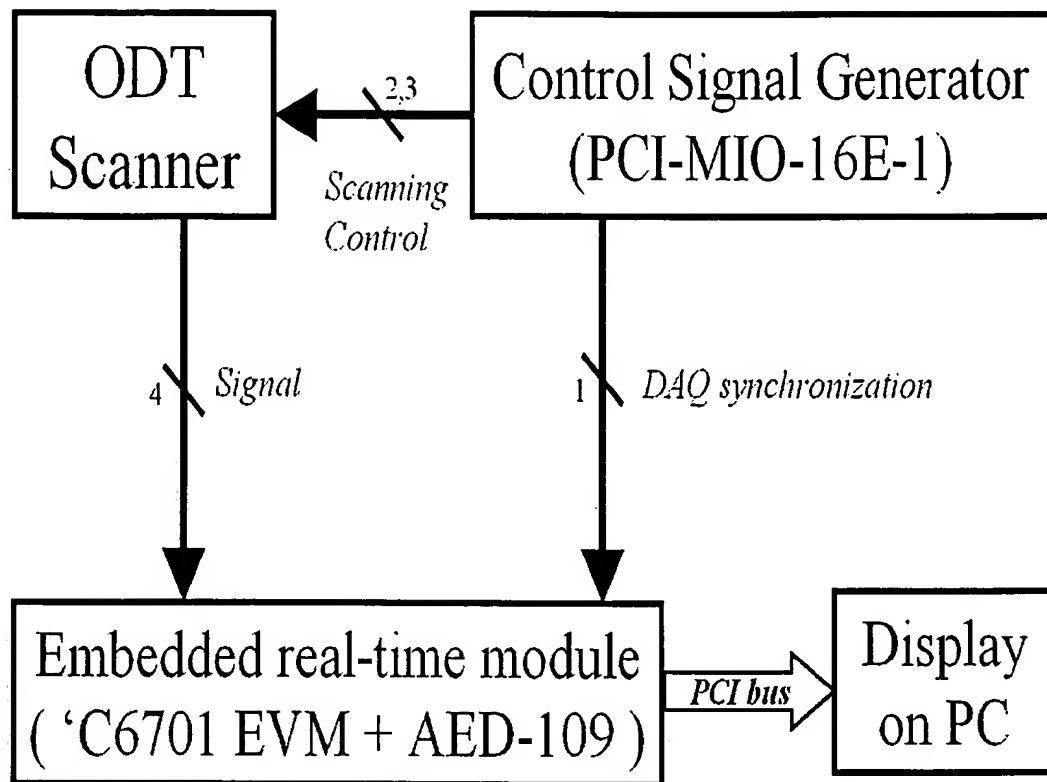


Fig. 2

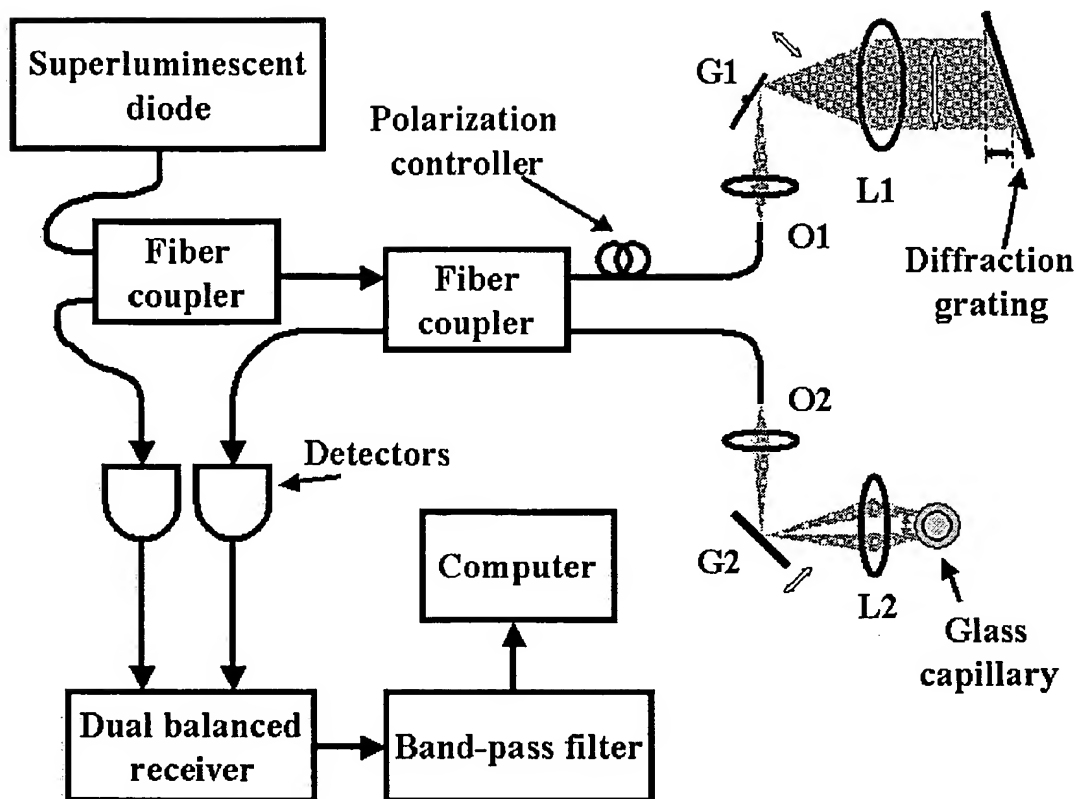


Fig. 3

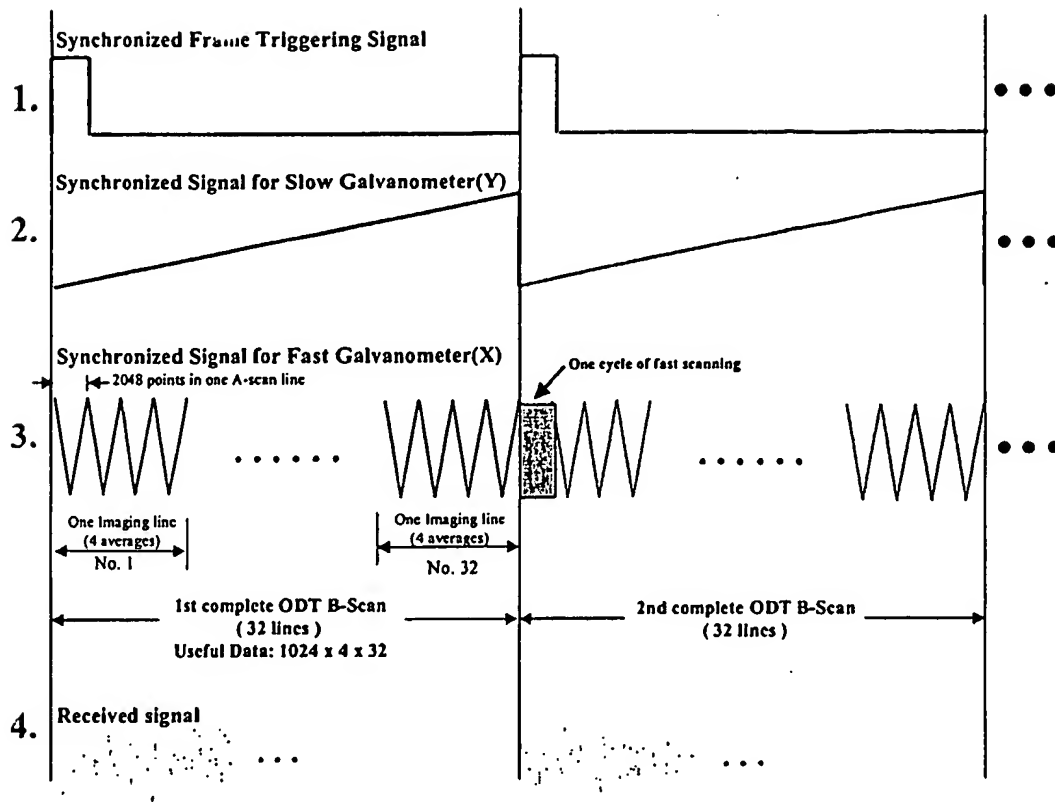


Fig. 4

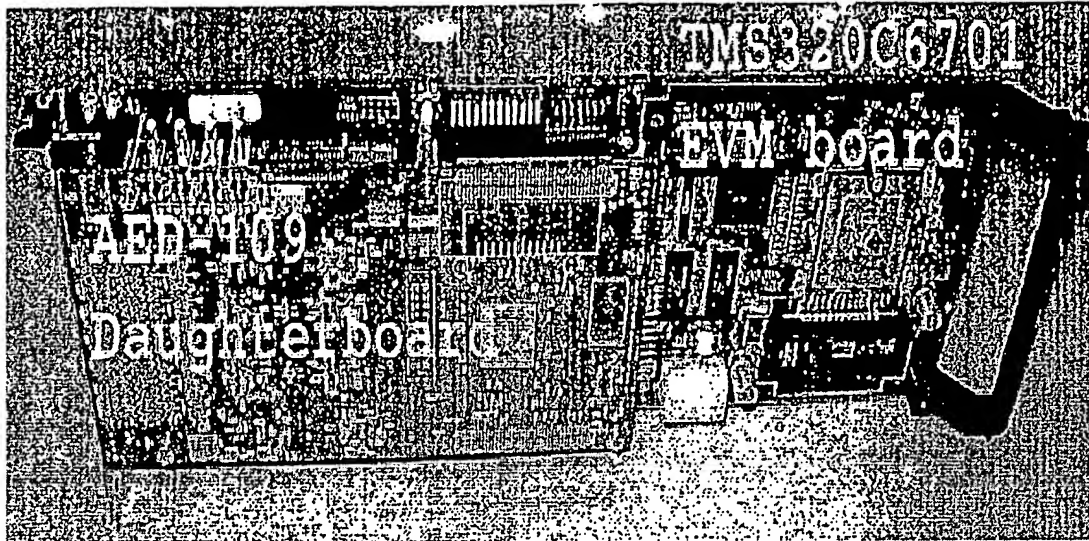


Fig. 5

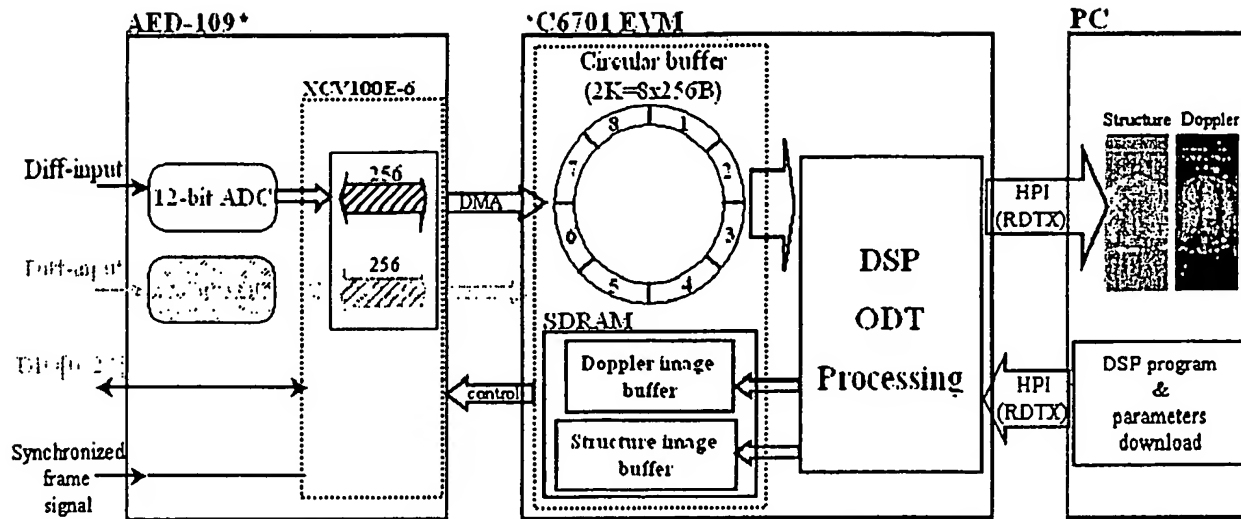
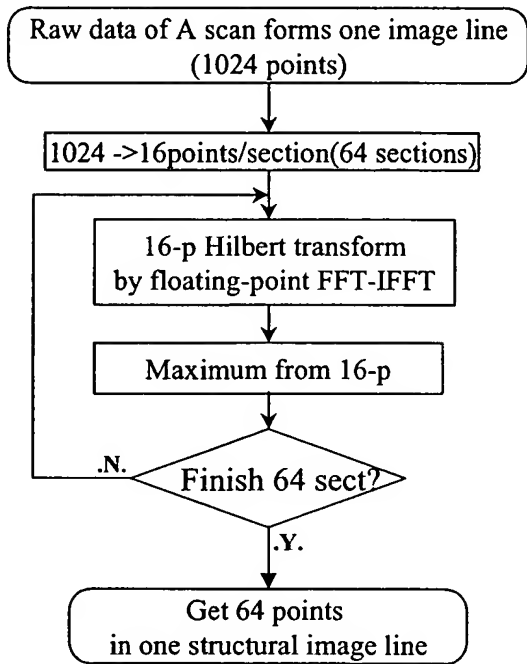


Fig. 6

10/524685

### Structural Image Processing Diagram



### Evaluation of DSP Calculation

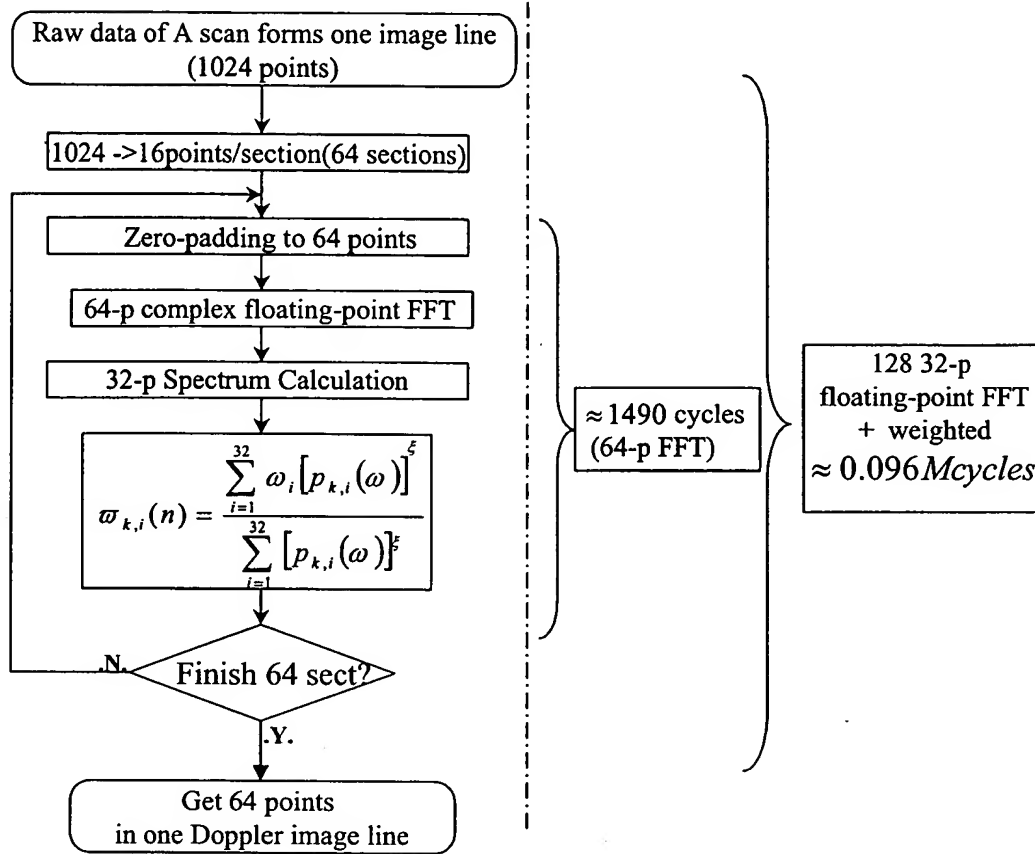
635 cycles  
16-p  
floating-point  
FFT-IFFT

$\approx 0.041 \text{ Mcycles}$

(a)

# Weighted Centroid Processing Diagram

# Evaluation of DSP Calculation

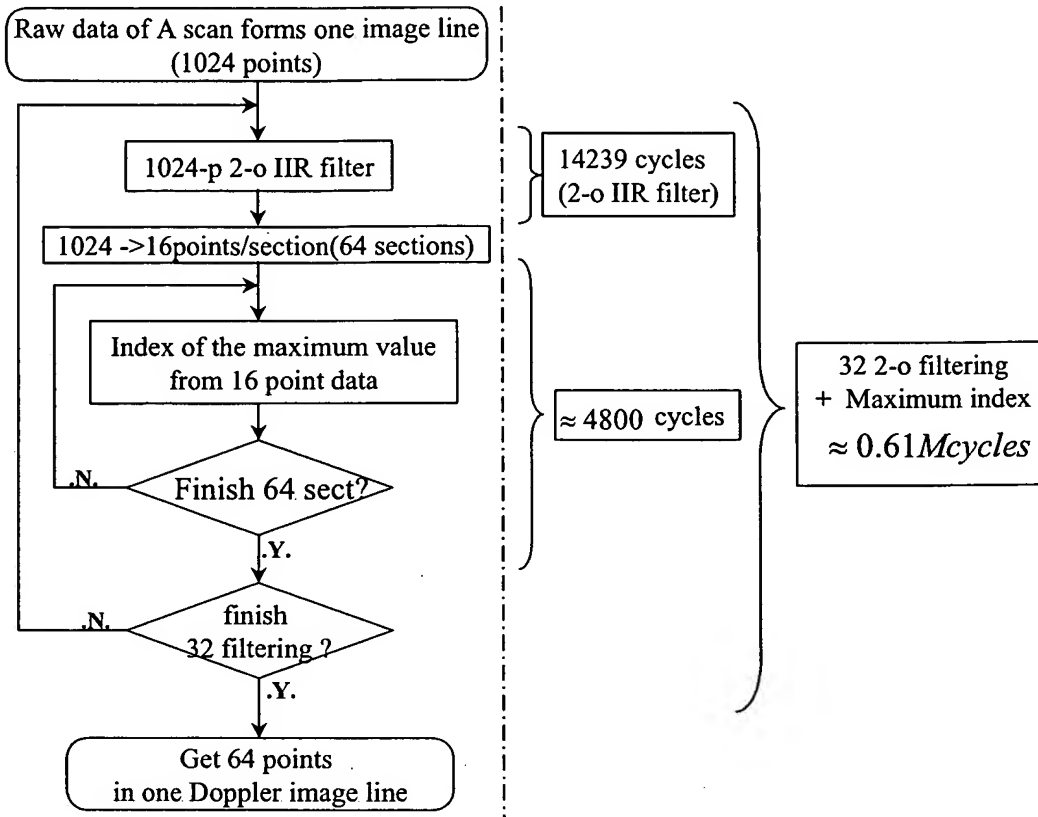


(b)



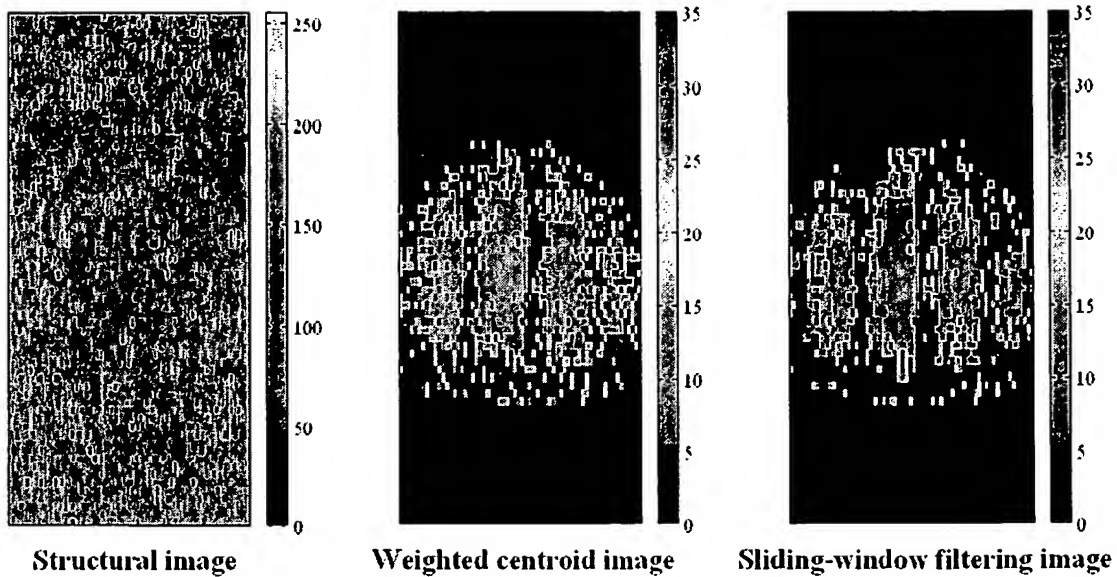
10/524685

# Sliding-window Filtering Processing Diagram Evaluation of DSP Calculation

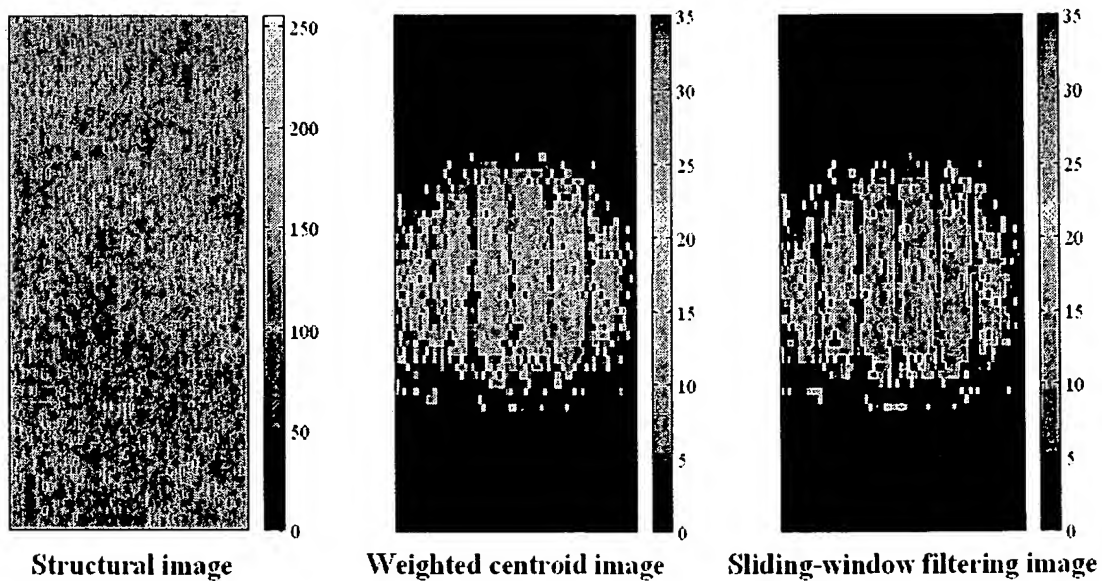


(c)

Fig. 7

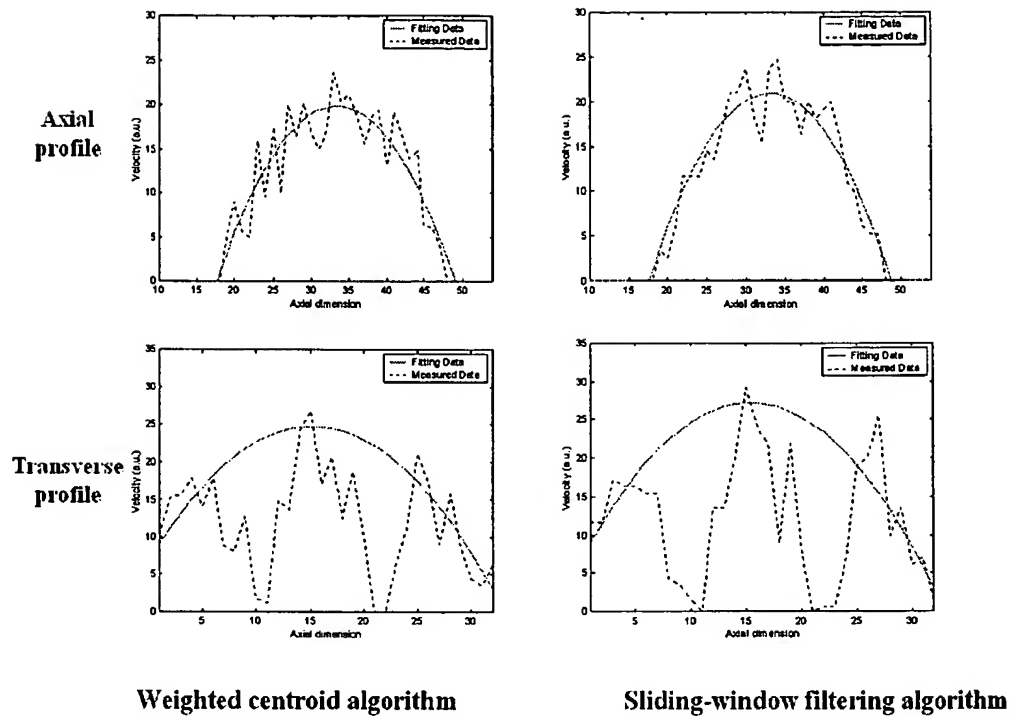


(a)

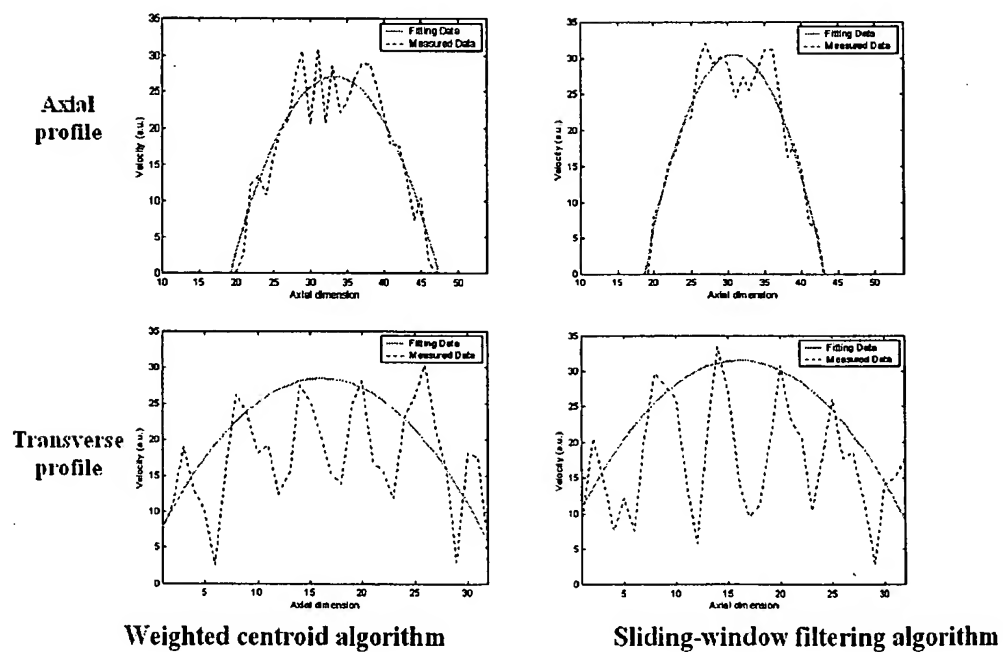


(b)

Fig. 8

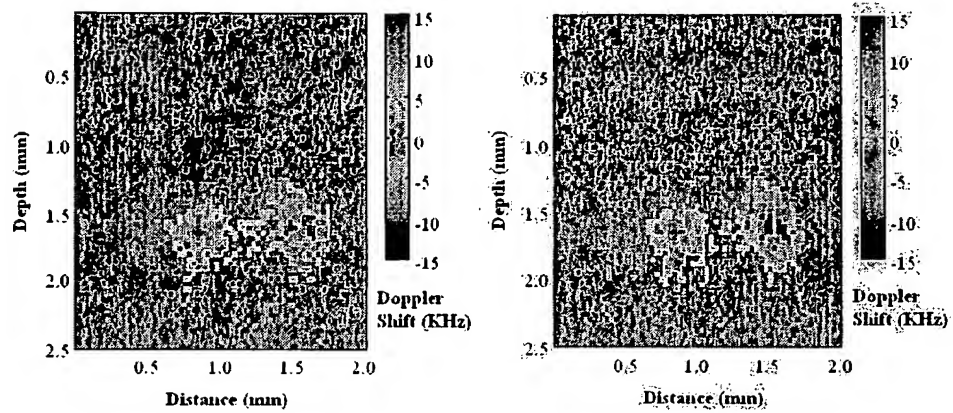


(a)



(b)

Fig. 9



Weighted centroid algorithm

Sliding-window filtering algorithm

(a)

(b)

Fig. 10

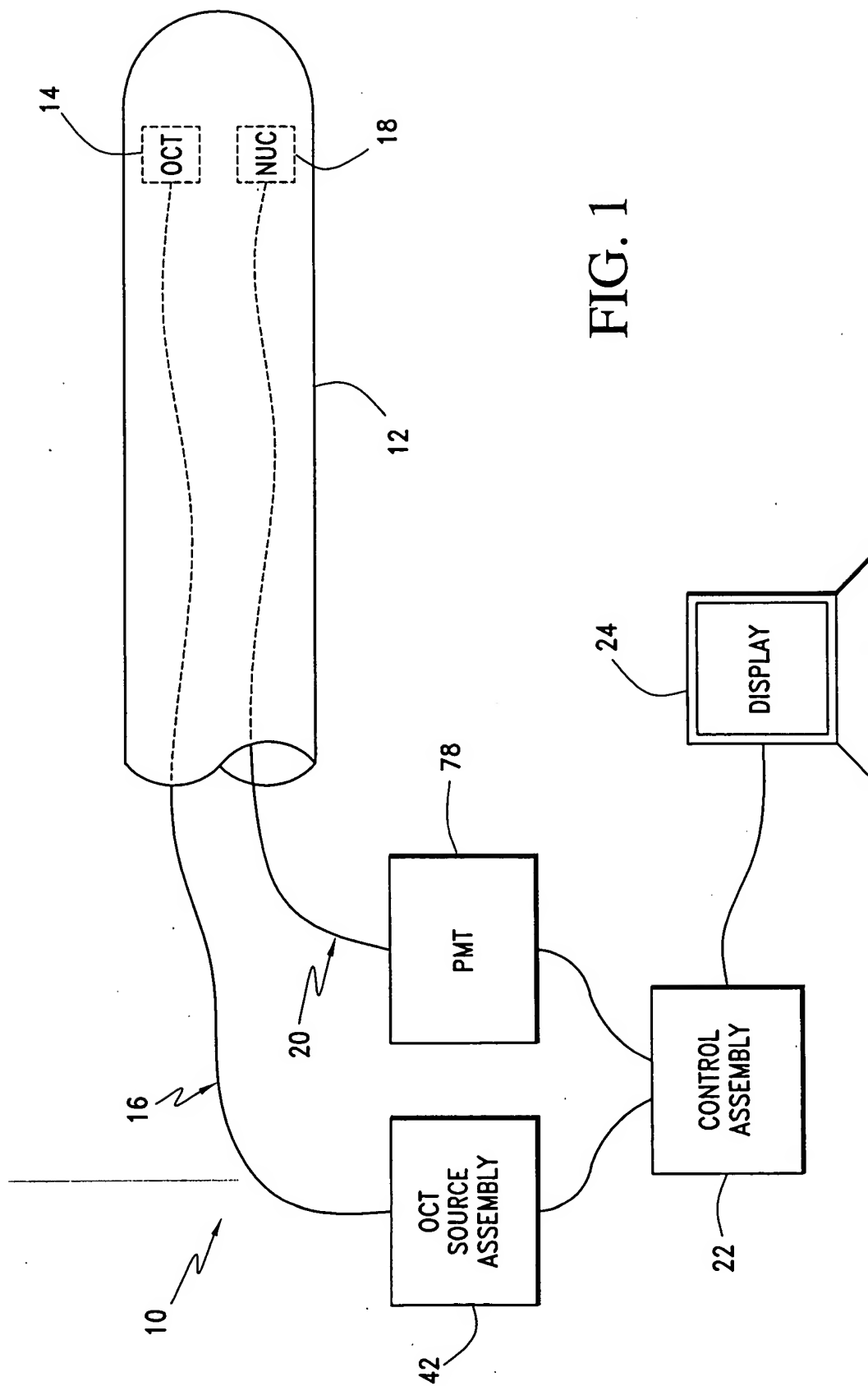


FIG. 1

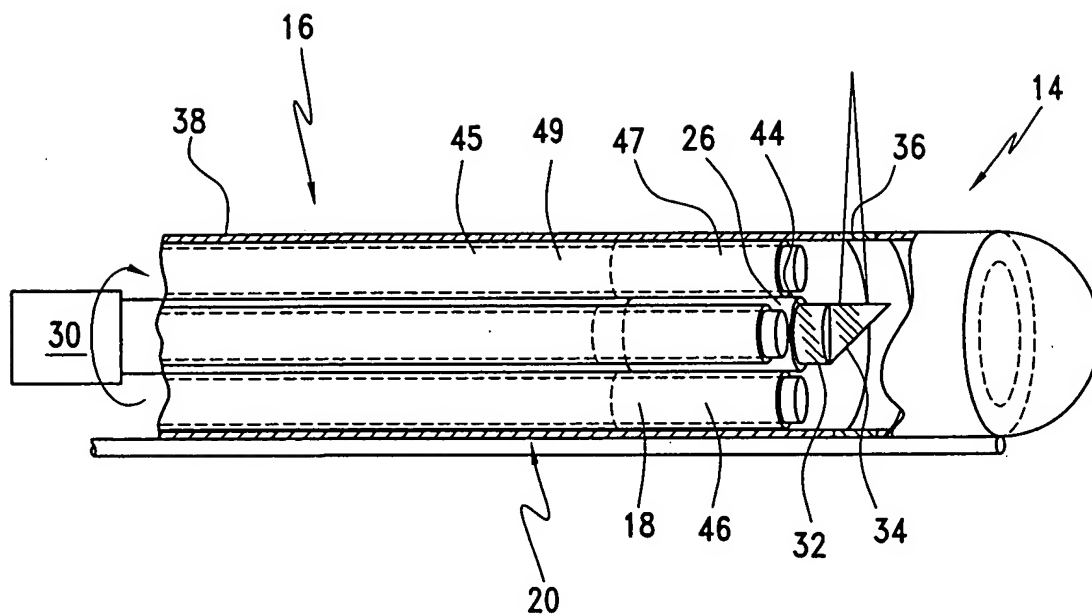


FIG. 2

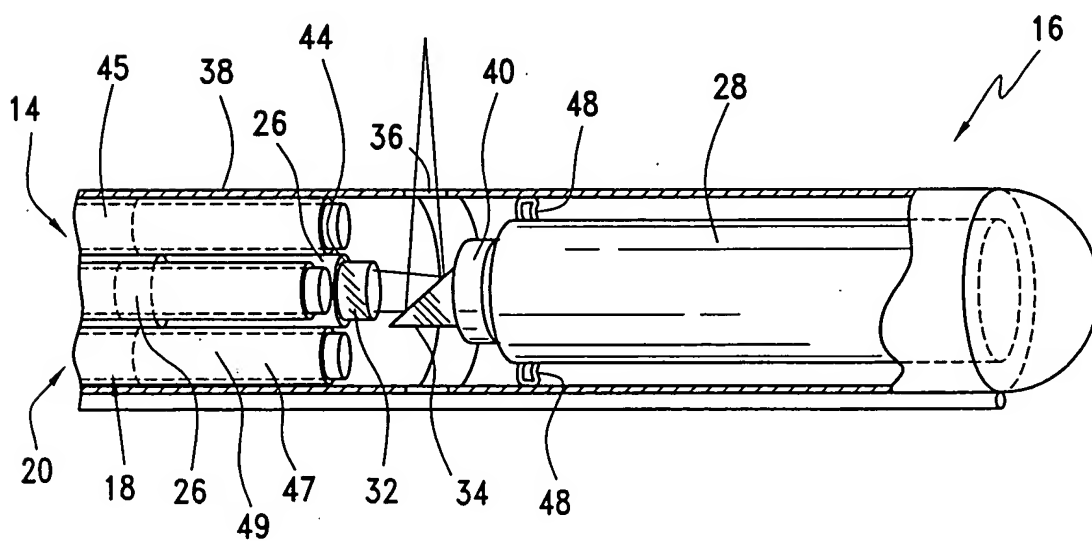


FIG. 3

3/11

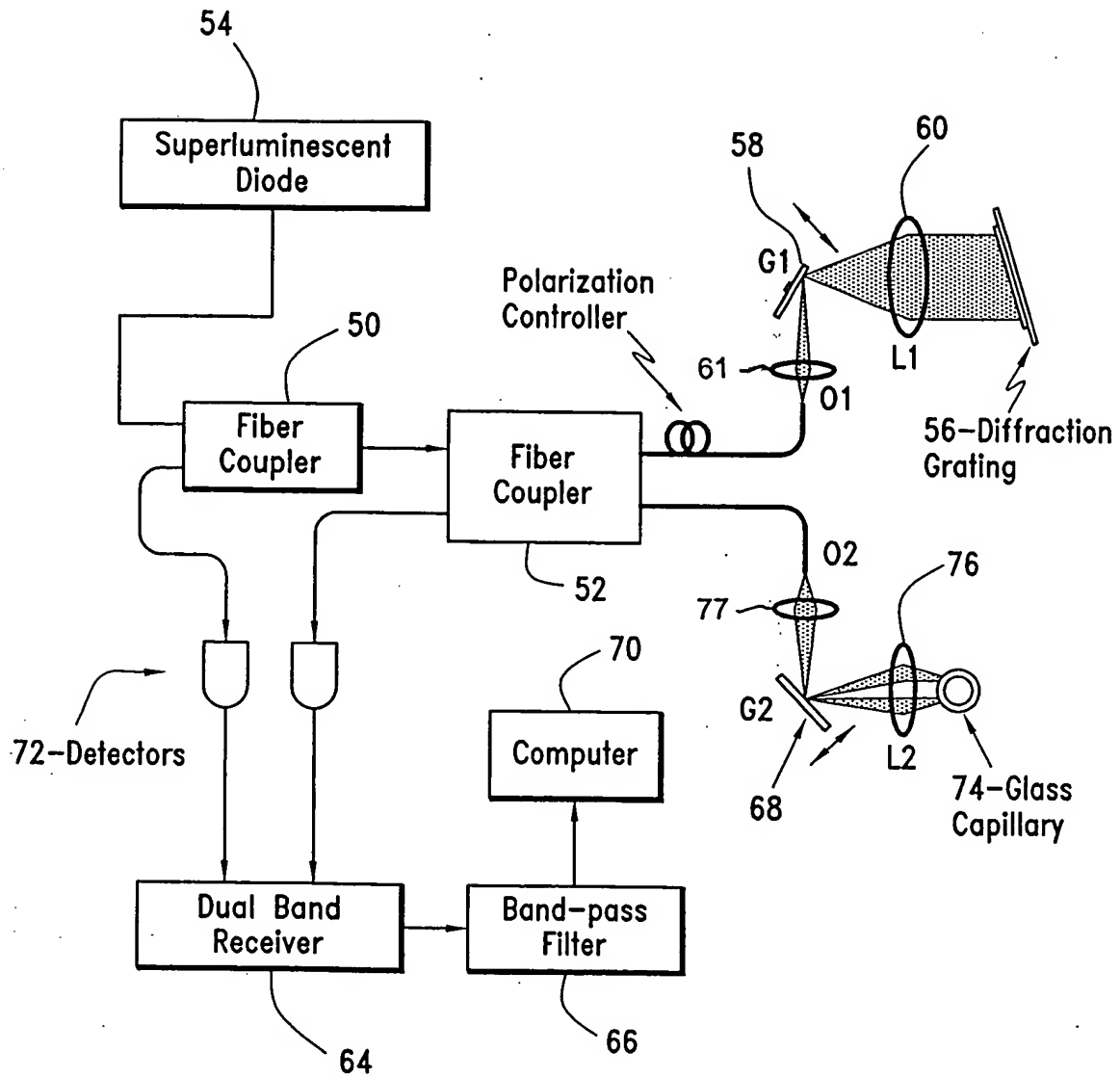


FIG. 4

4/11

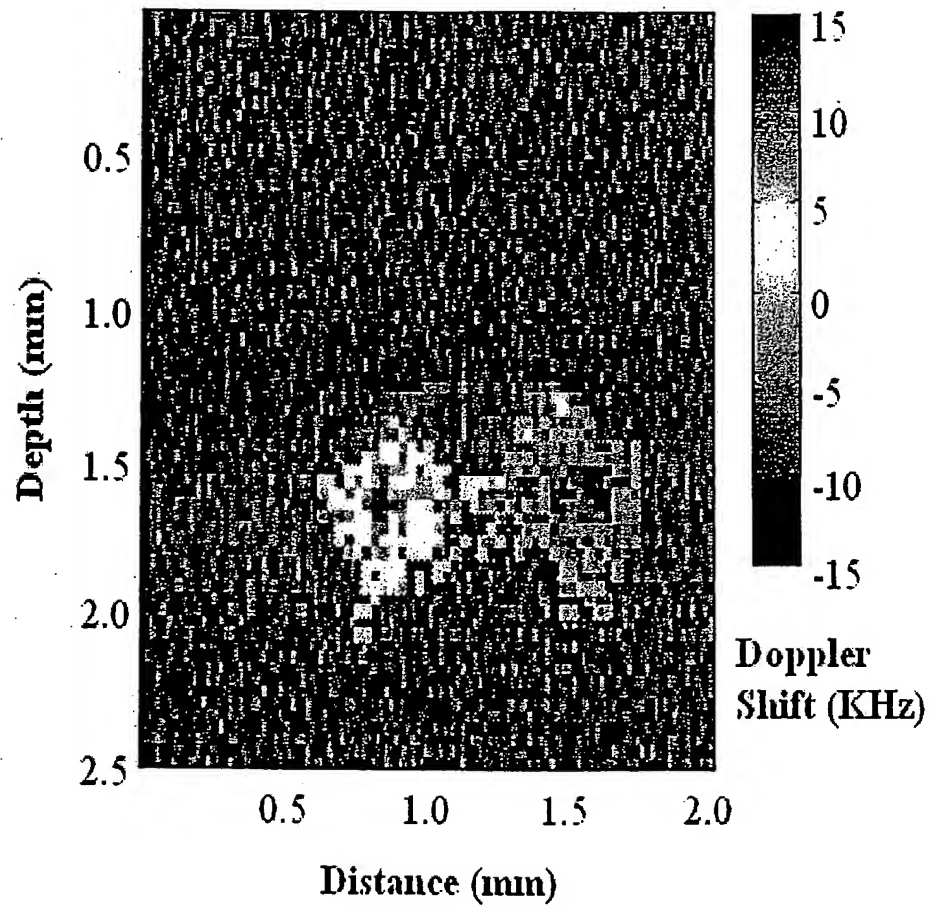


FIG. 5



5/11

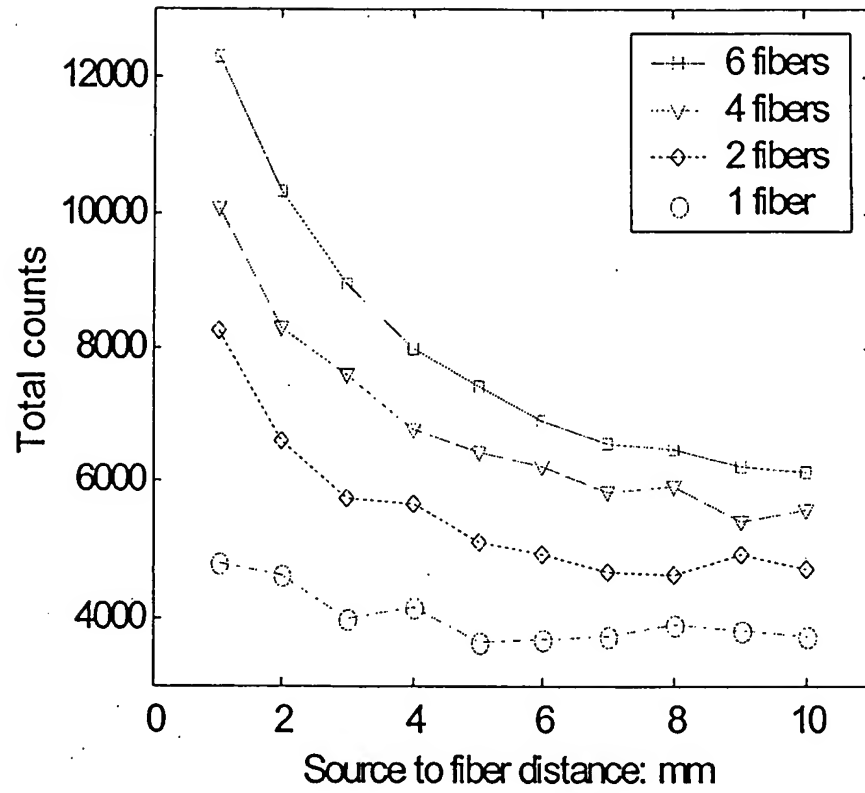


FIG. 6

6/11

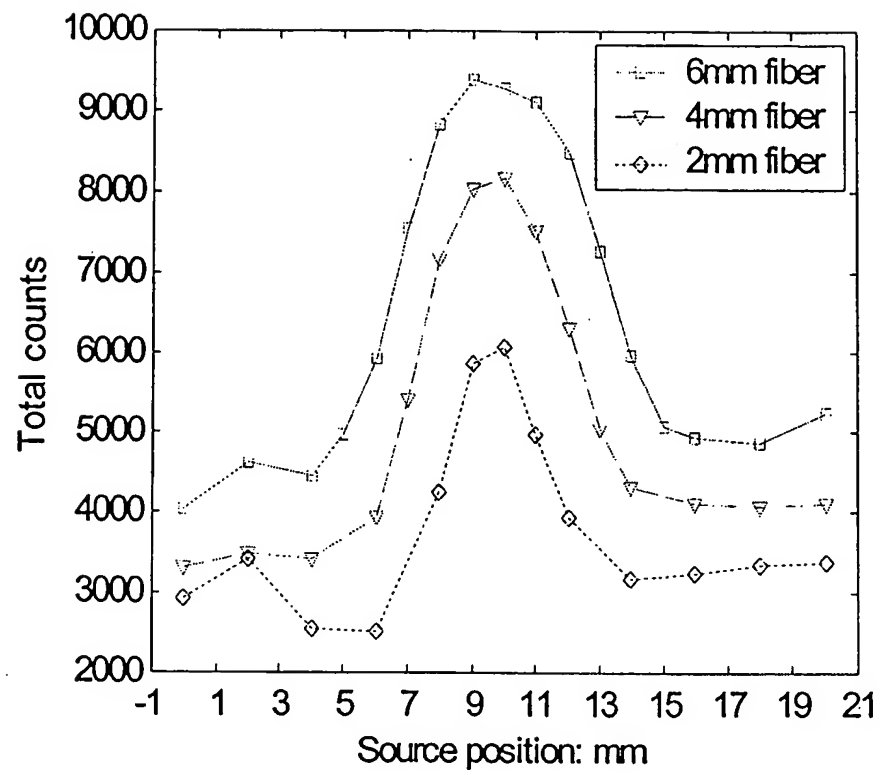


FIG. 7

7/11

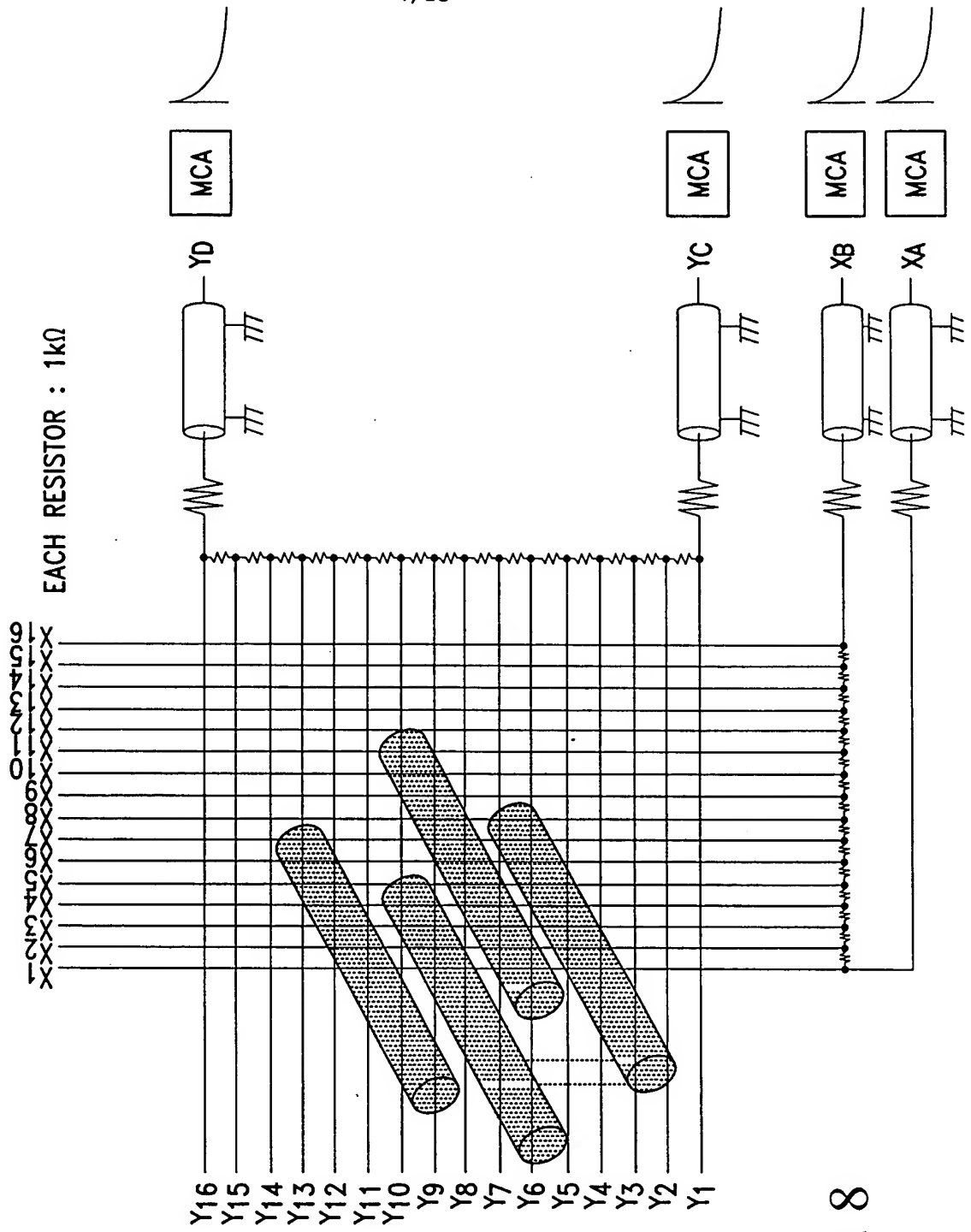
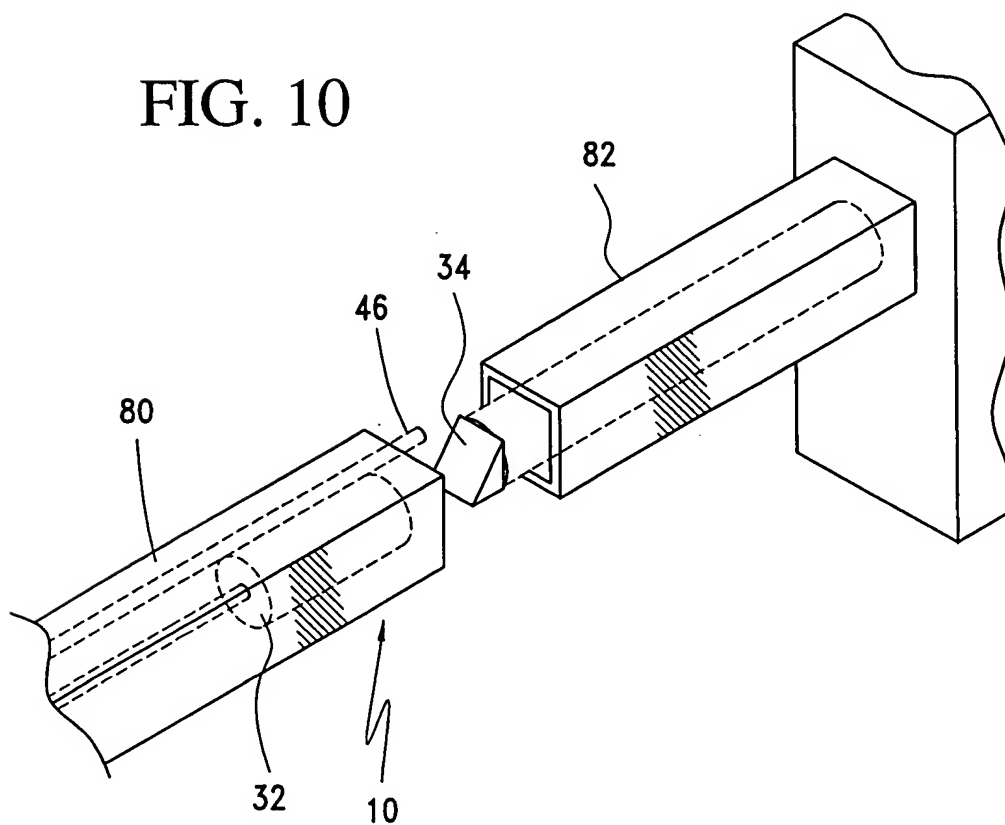
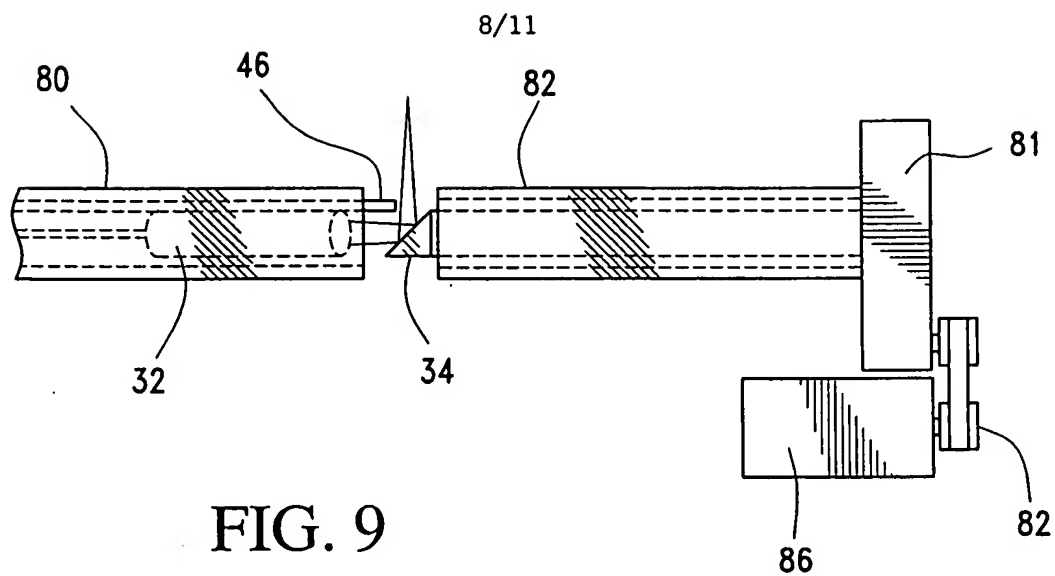


FIG. 8



10/524685

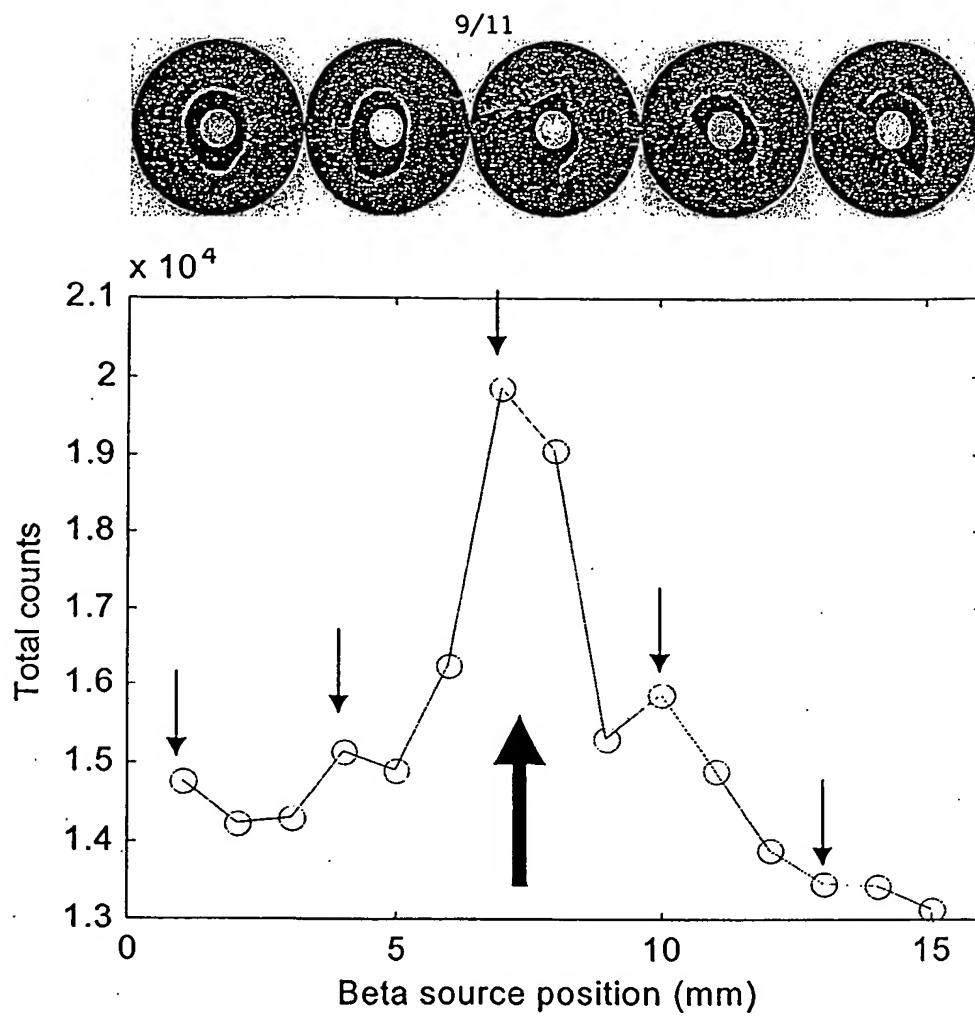


FIG. 11

10/524685

10/11

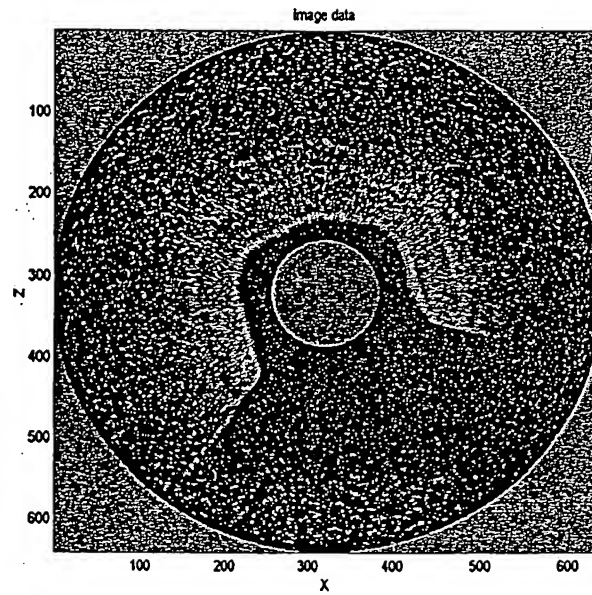


FIG. 12

11/11

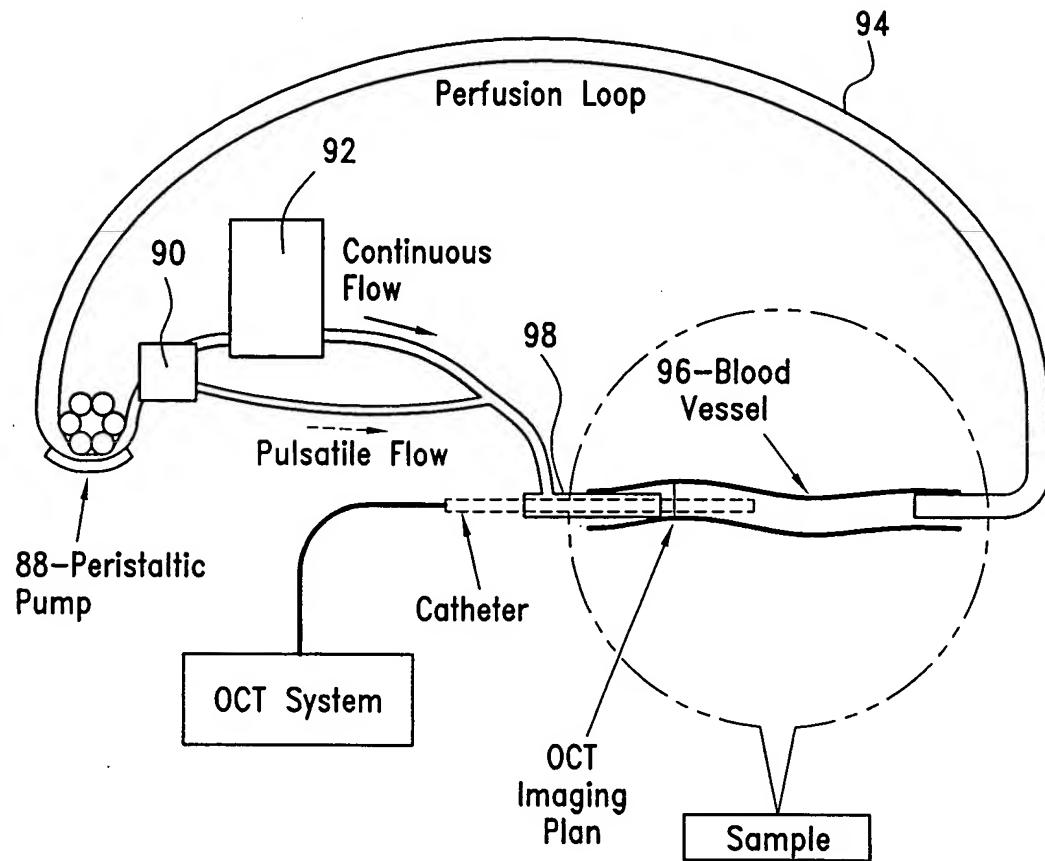


FIG. 13